Anycast Monitoring with passive TCP RTT

Giovane Moura (1) John Heidemann (2) Jeroen Bulten (3) Wes Hardaker (2) Joao Ceron (1)
Cristian Hesselman (1,4) Maarten Wullink (1) Marc Groeneweg (3) Marco Davids (1)
1: SIDN Labs 2: USC/ISI 3: SIDN 4: University of Twente

DNS operators strive to reduce clients' latency to authoritative servers, and often employ IP anycast [5, 7] with this goal. Large operators deploy dozen of globally distributed anycast *sites*, aiming at being physically closer to clients, ultimately reducing the latency to clients nearby.

Still, it is hard for an operator to know what latency its clients *experience*, given that operators usually do not have vantage points (VPs) within their clients' networks. As a result, they often rely on third-party services – such as Ripe Atlas [9, 10] and ThousandEyes [14] – which have their own set of VPs on various networks. The problem, however, these VPs are not necessarily located in the operator's most important client networks.

To overcome the issue with few VPs, a more comprehensive approach involving active measurements is Verfploeter [1], which consists in probing /24 IPv4 prefixes from the anycast address with ICMP and determing this way the RTT. However, Verfploeter does not scale well with IPv6.

We propose using DNS TCP RTT to estimate latency between authoritative servers and its *real* clients, as a complementary method to active measurement-based ones. TCP support is required on DNS servers [6], and latency can be estimated either during the TCP's session establishment or teardown. In our case, we measure DNS TCP RTT during the connection setup.

The advantages is that this method requires no extra measurements (passive data) and it allows for estimating latency from real clients, for both IPv4 and IPv6 (TCP latency estimation at endpoints has been used since 1996 [3], and it is widely used in passive analysis of HTTP [11], but has not been used for DNS).

Even though most of resolver-to-authoritative traffic is UDP (TCP covers 2-18% the Root DNS [4, 15]), it may cover networks where most queries come from – which his is the case for `.nl`. We recommend operators to determine if their DNS TCP traffic also covers their most important clients.

Case study: We present a case study in which we developed a authoritative server monitoring system that uses DNS TCP RTT. We call it Anteater and currently deploy it at SIDN [12], the `.nl` registry.

We show the system architecture in Figure 1. In this figure, we show one of its authoritative servers, with 5 sites ($A1$–$A5$). The first step consists in exporting `pcap` files from anycast sites to where it TCP RTT can be extracted. In the `.nl` case, we employ ENTRADA [13, 17], an open-source authoritative traffic data streaming warehouse, which extracts RTT from TCP queries [16] during the TCP session establishment.

Anteate retrieves periodically TCP RTT from ENTRADA, and aggregates it at different granularity levels:

(1) per authoritative servers and IP version
(2) per anycast site, authoritative server and IP version

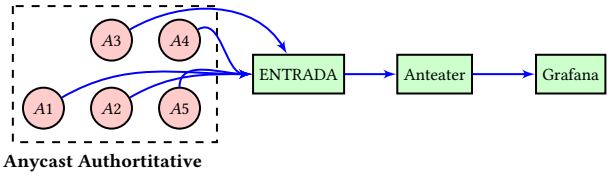

Figure 1: DNS TCP Latency Monitoring Architecture

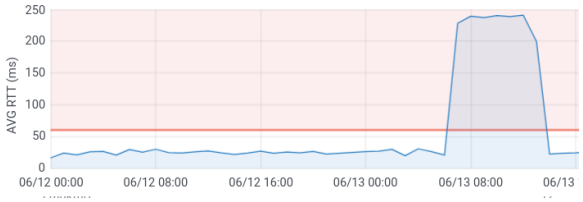

Figure 2: Johannesburg DNS TCP RTT

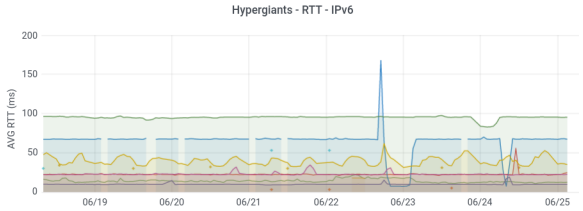

Figure 3: Hypergiant's IPv6 RTT to one of `.nl` auh servers

(3) per biggest client ASes in query volume and IP version – "hypergiants" [8] such as Google, Facebook, Amazon and Microsoft.

Anteater stores this information on a database (PostgreSQL), which we connect to Grafana [2]. Grafana is a interactive visualization web application, is used to plot the aggregated and that we have configured it to send alerts when RTTs go above chosen thresholds.

Real world example: By analyzing DNS TCP RTT, operators can know in near real-time the latency experience by clients. To illustrate that, consider Figure 2, in which we show the hourly average DNS TCP RTT for the Johannesburg site (IPv4) of one of the `.nl` authoritative servers. Around 8AM on on Jun 6th, the RTT experienced by this site's clients went from less than 50ms to more than 200ms. With this information, we contact the anycast operator, who could identify and solved the issue, which was a BGP misconfiguration in that site.

This example can be extended for other sites, server, and client ASes. Figure 3 shows the DNS TCP RTT for the hypergiants over IPv6 to one of the authoritative servers.

REFERENCES

[1] Wouter B. de Vries, Ricardo de O. Schmidt, Wes Haraker, John Heidemann, Pieter-Tjerk de Boer, and Aiko Pras. 2017. Verfploeter: Broad and Load-Aware Anycast Mapping. In *Proceedings of the ACM Internet Measurement Conference*. London, UK. https://doi.org/10.1145/3131365.3131371

[2] Grafana Labs. 2020. Grafana documentation. https://grafana.com/.

[3] Janey C. Hoe. 1996. Improving the Start-up Behavior of a Congestion Control Scheme for TCP. In *Proceedings of the ACM SIGCOMM Conference*. ACM, Stanford, CA, 270–280.

[4] ICANN. 2014. RSSAC002: RSSAC Advisory on Measurements of the Root Server System. https://www.icann.org/en/system/files/files/rssac-002-measurements-root-20nov14-en.pdf.

[5] D. McPherson, D. Oran, D. Thaler, and E. Osterweil. 2014. *Architectural Considerations of IP Anycast.* RFC 7094. IETF. http://tools.ietf.org/rfc/rfc7094.txt

[6] P.V. Mockapetris. 1987. *Domain names - concepts and facilities.* RFC 1034. IETF. http://tools.ietf.org/rfc/rfc1034.txt

[7] C. Partridge, T. Mendez, and W. Milliken. 1993. *Host Anycasting Service.* RFC 1546. IETF. http://tools.ietf.org/rfc/rfc1546.txt

[8] Enric Pujol, Ingmar Poese, Johannes Zerwas, Georgios Smaragdakis, and Anja Feldmann. 2019. Steering hyper-giants' traffic at scale. In *Proceedings of the 15th International Conference on Emerging Networking Experiments And Technologies*. 82–95.

[9] RIPE NCC Staff. 2015. RIPE Atlas: A Global Internet Measurement Network. *Internet Protocol Journal (IPJ)* 18, 3 (Sep 2015), 2–26.

[10] RIPE Network Coordination Centre. 2015. RIPE Atlas. https://atlas.ripe.net.

[11] Brandon Schlinker, Italo Cunha, Yi-Ching Chiu, Srikanth Sundaresan, and Ethan Katz-Bassett. 2019. Internet Performance from Facebook's Edge. In *Proceedings of the Internet Measurement Conference (IMC '19)*. ACM, New York, NY, USA, 179–194. https://doi.org/10.1145/3355369.3355567

[12] SID. 2020. Your World. Our domain. https://www.sidn.nl/en.

[13] SIDN Labs. 2020. ENTRADA - DNS Big Data Analytics. https://entrada.sidnlabs.nl/.

[14] ThousandEyes. 2020. Digital Experience Monitoring. https://www.thousandeyes.com/.

[15] Duane Wessels. 2020. RSSAC002-data. https://github.com/rssac-caucus/RSSAC002-data/.

[16] Maarten Wullink. 2019. ENTRADA 2.0 is here. https://www.sidnlabs.nl/en/news-and-blogs/entrada-2-0-is-here.

[17] Maarten Wullink, Giovane CM Moura, Moritz Müller, and Cristian Hesselman. 2016. ENTRADA: A high-performance network traffic data streaming warehouse. In *Network Operations and Management Symposium (NOMS), 2016 IEEE/IFIP*. IEEE, 913–918.